EFFICIENT IMPLEMENTATION OF CASCADED BIQUADS

TECHNICAL FIELD OF THE INVENTION

[0001] The technical field of this invention is digital signal processing, and more particularly to infinite impulse response filters.

BACKGROUND OF THE INVENTION

[0002] One of the most-used digital filter forms is the biquad. A biquad is a second order (two poles and two zeros) Infinite Impulse Response (IIR) filter. It is high enough order to be useful on its own, and because of the coefficient sensitivities in higher order filters the biquad is often used as the basic building block for more complex filters. For instance, a biquad low pass filter has a cutoff slope of 12 dB/octave, useful for tone controls; if a 24 dB/octave filter is needed, you can cascade two biquads and it will have less coefficient sensitivity problems than a single fourth-order design.

[0003] Biquads come in several forms. The most obvious, a direct implementation of the second order differential equation

$$(y[n]=a0*x[n]+a1*x[n-1]+a2*x[n-2]-b1*y[n-1]-b2*y[n-2]),$$

is called direct form **1** and is shown in FIG. **1**.

[0004] Direct form **1** is the best choice for implementation in a fixed point processor because it has a single summation point.

[0005] We can take direct form I and split it at the summation point as shown in FIG. **2**, and then take the two halves and swap them, so that the feedback half (the poles) comes first as shown in FIG. **3**. Now one pair of z delays is redundant, storing the same information as the other pair. Merging the two pairs yields the direct form II configuration shown in FIG. **4**.

[0006] In floating point applications, direct form II is preferred because it reduces memory requirements, and floating point computation is not sensitive to overflow in the way fixed point computations are.

[0007] We can improve on this configuration by transposing the filter. To transpose a filter, the signal flow direction is reversed. Output becomes input, distribution nodes become summers, and summers become nodes as shown in FIG. **5**. The characteristics of the filter are unchanged, but in this case the floating point characteristics are better. Floating point computation has better accuracy when intermediate sums are with closer values (adding small numbers to large number in floating point is less precise than with similar values).

SUMMARY OF THE INVENTION

[0008] An improved biquad filter is that is optimized for wide instruction word digital signal processors. The feedback path of the filter is modified, resulting in significant performance improvements.

BRIEF DESCRIPTION OF THE DRAWINGS

[0009] These and other aspects of this invention are illustrated in the drawings, in which:

[0010] FIG. **1** shows a direct form **1** biquad filter;

[0011] FIGS. **2** and **3** show intermediate forms of the biquad;

[0012] FIG. **4** shows a direct form **2** biquad filter;

[0013] FIG. **5** is a transposed form **2** biquad;

[0014] FIG. **6** illustrates an implementation of a biquad filter on a DSP;

[0015] FIG. **7** shows a modified biquad implementation; and

[0016] FIG. **8** shows a comparison of prior art and implementation according to this invention.

DETAILED DESCRIPTION OF PREFERRED EMBODIMENTS

[0017] FIG. **6** shows the transposed direct form II structure used in some implementations in Texas Instruments Digital Signal Processors (DSP). This implementation requires more than 10 cycles in the feedback path. Three to 6 cycles are used in addition block **601**, and 4 cycles in multipliers **602** and **603**. As shown in the figure, the feedback path to multipliers **602** and **603** originates at the output **604**.

[0018] FIG. **7** shows an improved implementation described in this invention. The feedback path to multipliers **702** and **703** originates from the output of storage element **701** instead of the output of summation block **704**. The coefficient in multiplier **706** is changed from b1 to b1+a1, and the coefficient in multiplier **707** is changed from b2 to b2+a2. This improvement results in requiring 7 cycles in the overall feedback path, 3 cycles in addition block **705** and 4 cycles in multipliers **702** and **703**.

[0019] FIG. **8** further demonstrates the implementation of this invention. The signal flow in the prior art is shown in table 1, and Table 2 shows the signal flow with the improved feedback path.

TABLE 1

```
out = in + d0
d0 = b1 * in + a1 * out + d1
d1 = b2 * in + a2 * out
```

TABLE 2

```
out = in + d0
t1 = (b1 + a1) * in + d1
t0 = a2 * d0
d0 = a1 * d0 + t1
d1 = (b2 + a2) * in + t0
```

4**[0020]** Table 3 shows performance benchmarks of the improved biquad filter executing on Texas Instruments C674x and C66x digital signal processors using single precision 32-bit floating point arithmetic, and Table 4 benchmarks filter performance using mixed/double precision floating point arithmetic on the same digital signal processors.